Scheme Daily 4

Due Sept. 1, 2023
CSE332 Programming Paradigms, Fall 2023

Overview Write a function `filterN` that returns only the numbers `n` through `m` from a `lat`. For example:

```
(filterN 4 6 (1 turkey 5 9 4 bacon 6 cheese))
    returns (5 4 6)
```

Recall that "numbers" are only positive integers, for now. Use only the functions `add1`, `sub1`, `and`, `or`, `<`, `>`, `null?`, `number?`, `zero?`, `define`, `lambda`, `cond`, and `else` in your code. You may also assume that `n` < `m`. As before, we provide a test script to get you started:

```
$ cp ~/esc-courses/fa23-cse-30332.01/public/scheme/d4/*.scm .
$ guile d4.scm
(1 turkey 5 9 4 bacon 6 cheese)
(4 4 4 1 1 bacon 9 9 8 6 6 6 1 4 5)
$
```

You will need to change the directory in the ice-9 import line in `d4.scm` to link to your dropbox folder.

Test cases and an example of the *correct* output is in the file `d4.scm`. Please implement your function inside this file and turn it in to the course dropbox, using subdirectories named `scheme/d4`. Remember to **include your name** in your `d4.scm` file. This homework will be graded out of 6 points.

Discussion The objective of this assignment is to practice the key strategy behind working with lists in functional programming: handle the first element of the list, then recurse on the remainder of the list. In other words, handle the `car` of the list, then recurse on the `cdr` of the list. In an iterative environment, usually the strategy is to use a loop. But in a *functional* environment, we use recursion. Think about the advantages and disadvantages to each strategy. One of the key advantages of the functional approach is that the number of requirements is much lower: the language only needs to support a small number of features. This is extremely useful when there certain types of hardware limitations.